25X1

Page Denied

THE THEORY OF ALGORITHMS

by

A.A. Markov

INTRODUCTION

1. In mathematics it is accepted to understand by "algorithm" the precise prescription of a definite calculational process leading from diversified initial data to the sought-for result.

A typical example of an algorithm is the Euclidean Algorithm finding the greatest common divisor of two natural numbers. An arbitrary pair of natural numbers plays here the role of initial values; the prescription consists of a sequence made up of a decreasing series of numbers, of which the first is the largest of the two given numbers, the second the smallest, the third is obtained as the remainder of the division of the first by the second, the fourth as the remainder of the division of the second by the third, etc. As long as there is no exact division without remainder, then the divisor of the last division is the desired result of the algorithm -- the greatest common divisor of two given natural numbers.

The following three characteristic features of algorithms determine their role in mathematics:

a) Precision of prescription, nothing left to arbitrariness, its obviousness definiteness of the algorithm;

b) the possibility of starting out with diversified and known bounded initial data -- the generality of the algorithm;

c) the directivity of the algorithm into obtaining of the particular desired result, in the long run obtained for proper initial data -- the conclusiveness of the algorithm.

2. The description of algorithms just proposed does not pretend to mathematical preciseness. In recent times, however, mathematicians have contented themselves with several indistinct notions which correspond to such a description. This was assumed as soon as the term "algorithm" was met in mathematics only in positive expressions of the following type: "for solution of such and such a problem one has an algorithm and it consists of the following." No negative results, no theorems of the impossibility of algorithms were to be demonstrated at this stage on the strength of the lack of precision of the notion of algorithm.

3. In recent times a sequence of authors have been elaborating theories leading naturally to a more precise notion of an algorithm. We have in view the works of Kleene in the theory of recursive functions [21, 23], the works of Church in the theory of λ- conversion [14, 15, 17], the work of Turing in the theories of computable numbers [31] and the work of Post in the theory of "finite conbinatory processes". [25]

This gave the possibility of setting up a sequence of important negative results -- a theory of impossibility of algorithms. From here come almost all the theorems of Church [16] about the unsolvability of common "decision problems" of the predicate calculus. In the character of major concrete results it is possible to produce a demonstration of the undecidability of a sequence of problems of the general theory of associative systems [3, 5, 6, 7, 10, 20, 28] and theories of integral matrices [4, 8, 10]. In recent times the most remarkable results of this kind are possibly the publication of P.S. Novikov on the undecidability of problems of the general theory of groups [12].

4. All such above mentioned negative results have fundamental values, for the further development of mathematics, in so far as they show the problematical dangerous potentialities of blind alleys, thus preventing the possibility of going into them.

The whole values for mathematics of the precision of the notion of algorithm is made apparent, however, in connection with the problems of a constructive basis for mathematics. As a basis for the precision of the notion of algorithm a definite constructive validity of arithmetic expressions can be given; for them the basis can be the design and construction of mathematical logic -- a constructive calculus of expressions and a constructive calculus of predicates. Finally, the last field of expression of the preciseness of the notion of algorithm is undoubtedly going to be constructive analysis -- the constructive theory of computable numbers and functions of computable variables now being found in a state of intensive investigation.

5. All of the above mentioned theories in point 3 are sufficiently complicated in themselves, and they lead to the precision of the notion of algorithm in an indirect way.

The theory of recursive functions [21, 23, 24] at the bottom deals with the particular occurence of an algorithm, when the natural number play the role of the initial variable, and the result of its application is a number. The translation into this general result requires for this the arithmetization of the initial data and of the results, which is arrived at by means of one or another "Godel enumeration" [18]- a process consisting of the application of certain very definite, but sufficiently complex algorithms.

The precision of the theory of algorithms fundamentally requires the theories of λ-conversion of Church, apart from the Godel enumeration, still a cumbersome formal appartus.

The theory of "computable numbers" of Turing, fundamentally undecidable in a constructive manner in the notion of computable numbers, brings the interesting to us precise notion of algorithm in an indirect way. The account of this theory is given by its author [31]; the demonstration of Post [28], gives an incomplete account.

Finally, the theory of finite combinatory processes of Post, closely related to the theory of Turing, was not completely worked out and consists in the main point of one definition [25].

In view of all that has been said the author considers it expedient immediately to study the precision of the notion and to develop the general theory of algorithms on the basis of this preciseness. Such a goal is pursued in the present monograph.*

The author thinks that he has succeeded satisfactorily in resolving the setting up of the problem and that the theory of algorithms being given here proceeds from a sufficiently simple and, together with that, ordinary difinition of "normal algorithm". In such a measure this pretension justifiedly is left to the reader to judge.

It was naturally easy to suppose that the theory being stated here of "normal algorithms" is equivalent to the theory of algorithms based on the notion of recursive function. It was shown by V.K. Detlovs that this was actually so [1].

*A short resume of the monograph are given by notes [9] and [10]. In paper [9] there occurs an annoying misprint: on page 185, in line 3 from the bottom, in place of α should be $\mathfrak{A}$ and in line 8 from the bottom in place of $\mathfrak{A}$ ($\mathfrak{A}$ should be α [illegible]) on page 188 in line 17 in place of the period should be a comma.

The present monograph is dedicated only to an account of the basic theory of algorithms and its application to the demonstration of the undecidable sequence of algorithmic problems. Another application of this theory which is mentioned in point 4 merits a special book, which the author in the future hopes to write.

6. The book is divided into six chapters, designated by Roman numerals, the chapters are divided into sections, and the sections into paragraphs. In each paragraph assertions (i.e., theorems, lemmas, corollaries) are enumerated separately and formulas separately.

The numbers of formulas are written at their left and enclosed in parentheses. The number of the paragraph is repeated at the left before the number of the assertion - spearated by a period from the number of the assertion.

The complete reference to an assertion consists of the number of the chapter, the number of the section (preceded by the symbol "§"), the number of the paragraph, and the number of the operation. These numbers are separated one from the other by periods, and all symbols are enclosed in square brackets. For example, [I. §3.9.3] designates a reference to assertion 3 of paragraph 9 of section 3 of Chapter I. In reference to an assertion in the same chapter the chapter number is omitted, in reference to an expression in the same section the number of the section is omitted.

References to formulae are handled analogously, with only the difference that periods are not included in the parentheses with the number of the formula and that in reference to a formula in the same paragraph the number of the paragraph is omitted.

Certain references made simultaneously are enclosed in large square brackets and are separated from each other by commas.

7. The contents of the book were expounded several times by the author in the courses of lectures given at Leningrad University. As a result of the exchange of opinion with the listeners and the accumulation of experience, the statement of the subject was gradually perfected. The author wants with pleasure to thank all his listeners for their many critical comments, which played a large positive role. In particular the author wants to thank R.V. Petropavlovski and N.A. Shanin.

The supervision of the book was elaborately overseen by V.A. Zalgaller and N.A. Shanin, who made a sequence of valuable comments, having an affect on the final edited manuscript. The author gives then his highest thanks for their heavy labours.

The author cordially thanks the Publishing House of the Academy of Sciences of the USSR for its attentive, solicitous concern for the book.

25X1

CHAPTER I

Letters, Alphabets, Words

This chapter has an introductory, auxiliary character: In it are considered a sequence of notions — "letters", "alphabets", "words", "entries", etc. — playing an essential role in the theory of algorithms.

A part of the material of this chapter is not necessary for our nearest goal - the formulation of a definite notion of the normal algorithm [II, § 3], — but will be used significantly later on [III, § 7, IV, § 3, IV. § 4, V, VI]. With this are included §§ 5 and 6, as well as lemmae § 4.2.2 - § 4.2.6, § 4.5.2 — § 4.5.8. To the reader desiring to get more quickly into the heart of things — into "normal algorithms" — we advise passing over the entire beginning of this material, to which he can later return for the purpose of completeness.

1. § Letters

1. By letters we will mean symbols which we will consider in their given application only as a whole.

In writing a book, the parts of the symbol a do not interest us, for example, but only the entirety of the symbol. In this sense typographical symbols are letters.

It is necessary to underline the conditional nature of this notion and the dependence of its range on an agreed understanding. For example, the symbol a' might be considered both as a letter, and as a symbol consisting of two parts a and ', according to adoption of this form by agreement.

2. In considering two given letters, we will state that they are either identical or different. For example, the first and sixth letters of the word "identical" are identical, and the first and sixth letters of the word "different" are different.

The notion of identity and difference of letters is also conditional. In particular in the identity of printed letters, much more rigid requirements are ordinarily presented than with the identity of letters written by hand; the identity of the first is nearer to a gometrical "equality" than the identity of the second. The conditionality of identity is shown especially sharply in establishing the identity of printed letters with letters written by hand.

The possibility of establishment of the identity of letters allows us, by the method of abstractive identification*, to set up the notion of abstract letters. The application of this abstraction consists in the given case in that we will begin to talk of two identical letters as one and the same letter.

For example in place of what would be said, that in the word "identical" two letters enter which are identical with "i", we shall say, "the letter 'i' enters twice in the word 'identical'." We here set up the notion of the abstract letter 'i' and shall consider the concrete letters identical with 'i' as representatives of this <u>one</u> abstract letter. <u>Abstract letters</u> are those letters which are considered the same to within identity.

Just as the application of abstractive identification is justified here, so is the condition observed; every letter being considered is identical with itself (reflexivity of identity); if one letter is identical with another, then the latter is identical with the first (symmetry of identity); two letters, identical with a third, are identical one to the other (transitivity of identity). In the following, idealizing several circumstances, we will consider these conditions strictly holding.

Developing abstractive identification in relation to a letter, we consider concrete letters as representatives corresponding to the abstract letter. Two concrete letters then and only then represent one and the same abstract letter when they are identical. In other words, the identity of abstract letters will be expressed by the identification of their representatives.

§2. Alphabets

1. For every application of letters we deal with a certain not very large set of abstract letters. It is impossible to use very large sets of abstract letters because of the amount of labor arising in establishing the identity and difference in the letters. In the natural organization

---

*We mean by this that which is ordinarily accepted to be called merely abstraction, the formation of an abstract notion by means of unification, identification of objects, connected by a relation of the type of equality, by means of correspondence (abstraction) from all differences of such objects. Our terminology represents for us the most complete because it is applied in contemporary mathematics to other types of abstractions, that is, correspondences (see, in particular, page 15 of the original, i.e. §3.4 and 3.5.)

of the size of letters and the large number of their forms certainly differences occur, for with difficulty the letters are distinguished--the identity and difference of the letters lose their legibility.

In place of this, it is impossible to indicate quantitative bounds for legibility for practical application of sets of abstract letters, that is, such a number N that sets of N abstract letters possess practical applications and that such sets of larger numbers of abstract letters are impossible. We underline, however, that for every application of letters we must deal with a certain <u>finite</u> set of abstract letters. This finite set may be given in the form of a list of concrete letters, representing the abstract letters of the set. Lists of such a kind we will call <u>concrete alphabets.</u>

Without the existence of a limited community, it is possible to impose in the concrete alphabet being considered the condition of absence of repetitions, consisting in that, any two letters appearing in the concrete alphabet are different. Further we always will suppose the condition to hold, without stating it.

2. For relations in the concrete alphabet it is also pertinent to set up an abstract identification. We will in this case abstract from not only the concreteness of the representation of the abstract letters but also the order of their appearance in the concrete alphabet. Accordingly therefore we will agree to call the concrete alphabet A equivalent to the concrete alphabet B if every concrete letter that occurs in A is identical with a certain concrete letter occurring in B, and conversely. In other words, we will call A equivalent to B if every abstract letter represented by a letter occurring in A is represented also by a letter occurring in B, and <u>vice versa</u>.

Identifying equivalent concrete alphabets, speaking of two equivalent concrete alphabets as of one and the same alphabet, we arrive at the notion of an <u>abstract alphabet</u>. We will consider every concrete alphabet as representing a certain abstract alphabet. Two concrete alphabets then and only then will represent one and the same abstract alphabet when they are equivalent.

An abstract alphabet is in essence the same as a set of abstract letters. In the same way, every abstract alphabet A synonomously defines a set of abstract letters, and similarly a set of these abstract letters, representatives of which are met in some concrete alphabet, represents A. Every concrete set of abstract letters is defined in this sense by one and only one abstract alphabet.

In consideration of abstract letters and abstract alphabets we will ordinarily omit the auxiliary word "abstract", that is, will write merely "letters" instead of "abstract letters" and "alphabet" instead of "abstract alphabet". We will also, in speaking of representatives understand by that those that are being represented. For example, we will speak of the letter *, understanding by this the abstract letter being represented by the symbol *. In other words, we will carry out abstractive identification with respect to letters and alphabets without expressing this clearly, in correspondence with ordinary practice.

In those cases where we are led to consider concrete letters and concrete alphabets, we will express this clearly by making use of the auxiliary word "concrete".

3. We will call letters, representatives of which are met in the representative of the alphabet A, <u>letters of the alphabet</u> A.

Every alphabet may, consistent with the representation, be considered as a set of letters of the alphabet. Corresponding with this we speak of the letters of the alphabet A that they <u>represent</u> A.

4. For consideration of arbitrary letters and of arbitrary alphabets it is customary to make use of letters as symbols of these objects. We thus used the letters A and B for designating (concrete and abstract) alphabets. In the future we will designate alphabets by large Russian* letters, letters by small Greek letters.

In speaking of "the letter α", "the letter β", etc., we will understand the letters represented and not the letters α, β themselves, etc.

*<u>Translator's note</u>: We will use instead large Greek letters.

In order to avoid confusion we will suppose that neither small Greek letters nor large Russian* letters belong to the alphabet being considered, a limit, it is understood, which is inessential.

5. We will use the notation

$$\alpha \in A$$

for expressing that the letter α belongs to the alphabet A, the notation

$$\alpha \neg \in A$$

for expressing that the letter α does not belong to the alphabet A; the notation

$$A = B$$

for the expression of equivalence of the alphabets A and B.

6. We will suppose in the future that the symbols "$\{$", "$\}$", and ";" do not occur as letters in the alphabet under consideration. We will make use of these symbols in the following example to record the alphabet by means of construction of its representatives. Writing one after another in the same sequence the representatives of all letters of the alphabet A under consideration, we will separate them by commas and enclose all of them in curly brackets. This gives us one of the concrete alphabets representing A. Corresponding to our agreement, we will, in speaking of this concrete alphabet, understand the abstract alphabet A, and likewise consider this concrete alphabet as a record of the abstract alphabet A.

For example, $\{a, b, c\}$ is a concrete alphabet, representing the abstract alphabet consisting of the letters a, b, and c. Corresponding to out agreement, we will, however, understand by "the alphabet $\{a, b, c\}$" just that abstract alphabet, considering "$\{a, b, c\}$" as its record.

The following alphabets will play an essential role in the future:

$$A_0 = \{a, b\};$$
$$A_1 = \{a, b, c, d\};$$
$$A_2 = \{a, b, c, d, e\};$$
$$A_3 = \{a, b, c, d, e, f, g, h, i, j, k, l, m\};$$
$$Ш = \{|\};$$

*Greek letters - translator.

$$\Sigma = \{|, *\};$$
$$\Delta = \{|, -\};$$
$$M = \{|, -, *, \square\};$$
$$T = \{|, -, *, \square, б\}.$$

7. We will spaek of the alphabet B that it is an extension of the alphabet A if every letter of the alphabet A is a letter of the alphabet B.

For example, A_1 is an extension of A_0; A_2 is an extension of A_1; A_3 is an extension of A_2; Σ and Δ are extensions of Ξ; M is an extension of both Σ and Δ, and T is an extension of M.

Limiting consideration to those alphabets which do not contain the symbol $\subset$, we will use the notation

$$A \subset B$$

as an expression that the alphabet B is an extension of the alphabet A.

In place of writing

$$A \subset B, \ B \subset \Gamma$$

we can write shorter

$$A \subset B \subset \Gamma.$$

In an analogous fashion we will use the notation

$$A \subset B \subset \Gamma \subset \Delta$$
$$A \subset B \subset \Gamma \subset \Delta \subset E$$

etc.

We have in particular

$$A_0 \subset A_1 \subset A_2 \subset A_3,$$
$$\Xi \subset \Sigma \subset M \subset T$$
$$\Xi \subset \Delta \subset M.$$

The following assertions are evident.

7.1 <u>If $A \subset B \subset \Gamma$, then $A \subset \Gamma$</u>.

7.2. <u>Every alphabet is an extension of itself</u>.

We will speak of the extension B of the alphabet A that it is an <u>essential extension</u> of the alphabet A if it is not identical with A, that is, if A is not an extension of B.

8. Earlier we defined a concrete alphabet as a list of concrete letters. In agreement with current understanding of the word "list", something must always be written down. In the list of families residing in a certain house, there must be at least one family. If the house is empty, then the list of dwellers ordinarily does not exist. It is possible, however to set up in this case the list of inhabitants in the house in the form of a piece of paper on which is written down only the title "List of dwellers in house no. 3 on NN street", and further nothing else. The possibility of a similar class of "empty" lists is expediently assumed in a definition of a concrete alphabet, which we will do. This permits us to consider along with the non-empty alphabets, possessing at least one letter, the empty alphabet possessing no letters at all. In our notation this will be written as:

$$\{\ \}.$$

It is expedient to consider that every alphabet is an extension of the alphabet $\{\ \}$.

3. § Words

1. We will call a sequence of concrete letters, written one after the other a <u>concrete word</u>. If the letters making up a concrete work R are represented by letters of the alphabet A, then we shall say that R is <u>a concrete word in A</u>.

For example

algorithm

is a concrete word in the English alphabet.

It is evident that every concrete word in an alphabet A is a concrete word in every extension of this alphabet.

In writing a concrete word in a given alphabet we will present a series of requirements of distinctness. Because the successive sequence of concrete letters composing a word plays an essential role, it must be clear in this sense that for any two such concrete letters, it must be clearly evident which of them stands at left (precedes) and which at right (follows).

The beginning and end of a word must be clearly indicated. We will, to this end, use quotation marks in ordinary examples, which evidently is admissible if the quotation marks are not among the letters of the alphabets being considered. Quotation marks besides are not considered as composing parts of words, serve only for indicating their limits, and will often be omitted.

9. That alphabet of all letters contained in at least one of the alphabets A and B, we will designate as the <u>union</u> of the alphabets A and B; the alphabet of all letters contained in both those alphabets will be designated as their <u>intersection</u>. In an analogous manner we will define the union and intersection of three or more alphabets.

The alphabet consisting of the letters of the alphabet A, but not contained in the alphabet B, we will call the difference of the alphabets A and B.

Considering only those alphabets in which the symbols "$\cup$", "$\cap$", "$\setminus$" do not appear, we will use the notation

$$A \cup B$$

to designate the union of the alphabets A and B; the notation

$$A \cap B$$

for designating their intersection; the notation

$$A \setminus B$$

for designating their difference; and the notations

$$A \cup B \cup \Gamma$$

and

$$A \cap B \cap \Gamma$$

for corresponding use in designating union and intersection of the alphabets A, B, and Γ.

We have, evidently,

$$\Sigma \cup \Lambda \cup \{\square\} = M$$
$$\Sigma \cap \Lambda = \Xi$$

The use of quotation marks gives also the possibility of consideration of empty words of the form

" ",

containing no concrete letters.

We will consider every empty word as a word in every alphabet.

A very important future requirement will consist in the uniqueness of the decomposition of a word into representative letters of the given alphabet -- the requirement of impossibility of "variant reading".

This is by no means always observed. For instance, the concrete word

a'a

in the alphabet $\{a, a', 'a\}$ can be considered as a sequence of the two concrete letters representing the letters "a" and "'a" of this alphabet, or as a sequence of the two concrete letters representing its letters "a'" and "a".

It is necessary to exclude completely the possibility of such variant reading of words. This can be attained by imposing proper limits in the alphabets being considered and in the manner of writing the words in them.

It is possible, for example, to present the following two requirements:

a) Every letter of the alphabet must be joined, that is, must be formed without loss of contact of pen with paper;

b) in writing words an interval should be left between every two neighboring letters.

The first of these two requirements imposes limits on the alphabets being considered, and the second on the manner of writing words. It is clear that for observance of these requirements every concrete word in the alphabet being considered will be decomposed into concrete letters in a unique way.

It is possible to have other systems of requirements, guaranteeing this uniqueness. However, requirements a) and b) are convenient in this regard, that they admit unlimited construction of the union of the alphabets being considered: the union of any two alphabets satisfying requirement a) also satisfies that requirement. Together with this requirement a) does not admit any essential limitations on the alphabet, because for every disconnected letter of the alphabet, evidently, it is always possible to substitute a connected letter, different from all the remaining ones.* Henceforth we will understand by

*We remark, however, that the Russian printed alphabet does not satisfy condition a) because of the disconnected letter "bI". Nevertheless, decomposition of a word in this alphabet into letters is unique. This shows that conditions a) and b) are by no means necessary for unique written words.

"alphabet", an alphabet satisfying condition a), by a "concrete word" a concrete word written or printed with observance of condition b), by "letter" a connected letter.

In the future we will designate words (concrete as well as abstract) by capital English letters, understanding that these letters are not letters of the alphabet being considered. In speaking of "the word P," etc., we will always have in mind the word designated by the letter P (and not the word composed of the single letter P).

2. We will say of the concrete words P and Q that they are equal if they consist of the identical concrete letters, identically ordered. Two empty words we will besides also call equal, and an empty and a non-empty word will be called unequal.

The concrete word set apart on a separate line on page 7, is equal to the concrete word "algorithm".

In ascertaining the equivalence of concrete words one can apply the following method.

Let two concrete words P and Q be given. If they both are empty, then they are equal. If one is empty and the other not, then they are not equal. If neither one nor the other of the two words is empty then we compare their first concrete letters. If they are not identical, then the words P and Q are not equal. If these letters are shown to be identical, then we transfer into consideration the concrete words P_1 and Q_1, obtained from P and Q as a result of cutting off the first concrete letter (that is, transferring the initial quotation mark across the first letter). P and Q are equal then and only then when P_1 and Q_1 are equal. Considering P_1 and Q_1 in the same fashion as we considered P and Q, we either establish their equivalence and thus the equality of P and Q; or we establish theirnon-equivalence, and thus P and Q are not equal; or finally we bring into consideration the words P_2 and Q_2, obtained from P_1 and Q_1 be means of cutting off their first letter. We deal with these as we dealt with P_1 and Q_1, etc. This process of sequential determination of the identity of the first concrete letters must in the end be cut short, because P_1 consists of one less concrete letter than P, P_2 of one less concrete letter than P_1, etc. It terminates in certain concrete words P_n and Q_n, for which it is [illegible] the initial words P and

The just described method of ascertaining the equality of concrete words evidently possesses the three requirements, which we recorded as characteristic of the features of algorithms [Introduction1]; it takes the form of an obvious order, not being laid down arbitrarily; it is possible to apply it to different initial data -- to every pair of concrete words in a given alphabet, it is directed towards obtaining a certain result, being given in the end by the correctness of the answer "yes" or "no" to the posed question of the equality of the words. Conserving for the time being the imprecise notation of algorithm, we indeed will call this method the algorithm for equality of words.

It is not difficult to see that the equality of concrete words obeys the laws of reflexivity, symmetry, and transitivity: every concrete word is equal to itself; if the concrete words P and Q are equal then the concrete words Q and P are equal, two concrete words equal to a third are equal to each other.

2.1 Every concrete word that is equal to a concrete word in the alphabet A is a word in A.

This follows immediately from the definition of concrete words in a given alphabet and the equality of concrete words.

We may now, applying abstractive identification, set forth the notion of abstract word. The application of this abstraction will consist in a given case in that we will speak of equal concrete words as of one and the same word.

For example, we will say that one and the same word appears on the separate line on page 7. This signifies that we have set up the notion of the abstract word "algorithm", and we are considering just that above-mentioned concrete word as a representative of that one abstract word.

The application of abstractive identification with respect to words was justified by the above mentioned laws of reflexivity, symmetry, and transitivity of the equivalence of words. It is connected with consideration of concrete words as representatives of abstract words. Two concrete words in this case then and only then represent one and the same abstract word when they are equal.

For expression of equality of concrete words, that is identity of the abstract words represented by them, we will apply the ordinary symbol of equality.

4. Two representatives of one and the same abstract word P consist of identical concrete letters, identically positioned. Abstract letters are represented by these concrete letters, one and the same for both representatives, and are defined, in this way, by the abstract word P. These abstract letters we will call *letters of the word P*.

Because we count any two empty concrete words as equivalent, we must consider empty concrete words as representatives of one abstract word – *the empty abstract word*.

The empty abstract word does not have letters.
Understanding that the symbol "Λ" is not a letter of the alphabet being considered, we will denote by this sign the empty abstract word.

We will say of the abstract word P that it is an *abstract word in the alphabet A*, if all letters of the word P are letters of the alphabet A. In other words, the abstract word P is considered an abstract word in the alphabet A if any (and thus every) representative of the word P is a concrete word in A.

We will consider the empty abstract word as an abstract word in every alphabet (even in the empty alphabet).

5. In the future in the consideration of alphabets, words, and algorithms, *abstraction of potential feasibility* will play an important role.

This consists in abstraction from the real boundaries of our constructive possibilities which stipulate the limitedness of our lives in space and in time. In application to alphabets this abstraction permits us to reason about the size of the alphabet, and, in particular to be sure that new letters can be adjoined to every alphabet. In the application to words we get in this way the possibility to reason about the length of words as about their feasibility. Their feasibility is potential; their representatives would be practically feasible, if our life would endure sufficiently long and we would have sufficient room and material for practical feasibility of these representatives. Adopting these abstractions, we will understand in the future simply by "word" the abstract potentially feasible word.

We will assume the possibility of reasoning about words (in this sense) completely, also, as we reasoned about the practical feasibility of words in which the essence of the abstraction of potential feasibility in a given application consists. In particular, one may speak of the letters of a word, of its representatives, that it is a word in a given alphabet, etc.

The abstraction of potential feasibility, as in abstractive identification, is completely necessary for mathematics. On these two abstractions are based, in particular, the notion of natural numbers.

6. Attaching on the right some representative of the word Q to some representative of the work P, we get a representative of a certain new word. The latter, evidently, does not depend on the given set of representatives of the words P and Q, that is it is completely defined by these words. The word, obtained in this way, resulting from the words P and Q, we will denote as the <u>union</u> of the words P and Q.

For instance, the word "input" is the union of the words "in" and "put".

The abstraction of potential feasibility gives the possibility of considering the union of any two words.

6.1. <u>The union of any two words can be formed</u>.
The following assertion is evident.

6.2. The union of two words in an alphabet A is a word in A.

We will agree at present to designate by the symbol $\overline{PQ}$ the union of the words P and Q.

The associative law of union is evident

(1) $\overline{\overline{PQ}R} = \overline{P\overline{QR}}$

where P, Q, and R are any words.

This gives the possibility for operating with the union to leave out the upper lines, which we will hence forth do. Both parts of the equality (1) will be written then identically, in the form PQR. It is evident that PQR is the word, the representative of which is obtained as the result of writing a sequence at the beginning of which is a representative of the word P, then a certain representative of the word Q, and finally, a certain representative of a word R.

We will call the word PQR the union of the words P, Q, R; the word PQRS the union of the words P, Q, R, S, etc.

It is clear that

(2) $P\Lambda = P, \Lambda P = P$

for every word P.

In the future we will often have to deal with a sequence of words, designated by letters with numerical indices. The union of such words in a sequence of increasing indices will be designated in the following manner.

For $i > j$ the symbol

(3) $P_i \cdots P_j$

will always designate the empty word, independent of how the words P_i and P_j are defined. For $i = j$ this symbol will designate the word P_i, if that word is defined. For $i < j$ the symbol (3) will denote the union of the words P_i, P_{i+1}, etc. through P_j inclusive, if all these words are defined.

We have, in this way

(4) $P_i \cdots P_{i-1} = \Lambda$

(5) $P_i \cdots P_j = P_i \cdots P_{j-1}P_j$

(6) $= P_iP_{i+1} \cdots P_j$

in which the equalities (5) and (6) take place under the condition that $i \leq j$ and that all words P_k are defined, where $i \leq k \leq j$.

The union of several words designated by letters with indices will be designated in a sequence of decreasing indices analogously.

For $i < j$ the symbol

$P_i \cdots P_j$

will designate Λ; for $i = j$ it will designate P_i, if P_i is defined; for $i > j$ it will designate the union of the words P_i, P_{i-1}, etc., through P_j, inclusive, if all these words are defined.

We have in this way

$$P_i \cdots P_{i+1} = \Lambda$$

$$P_i \cdots P_j = P_iP_{i-1} \cdots P_j$$

$$= P_i \cdots P_{j+1}P_j$$

the latter under the condition that $i \geq j$ and that all words P_k are defined, where $i \geq k \geq j$.

7. A concrete word, that is a sequence of concrete letters, can in particular, be one concrete letter. The abstract word representing it is then an abstract letter. In this way, every letter is a word, and in particular, every letter of the alphabet A is a word in A.

Every non-empty word in the alphabet A is, evidently, either a letter of this alphabet, or the union of several of these letters; that is, the following assertion holds.

7.1 <u>Every non-empty word in the alphabet A is represented in the form</u>

(1) $\xi_1 \ldots \xi_k$,

<u>where $k > 0$ and $\xi_1, \ldots, \xi_k$ are letters of the alphabet A.</u>

In view of requirements 1a) and 1b), guaranteeing the impossibility of "variant reading", the representation of word in the alphabet A in the form (1), where $\xi_1, \ldots, \xi_k \in A$, is unique, which is expressed by the following assertion.

7.2. <u>If $\xi_1 \ldots \xi_k = \eta_1 \ldots \eta_\ell$, where $k \geq 0$, $\ell \geq 0$ and $\xi_1, \ldots, \xi_k$, $\eta_1, \ldots \eta_\ell \in A$, then $k = \ell$ and $\xi_i = \eta_i (1 \leq i \leq \ell)$.</u>

8. The number k, occuring in the representation (1) of the word defined consistent with 7.2 synonymously with this word, we will call the <u>length</u> of the word P and designate by the symbol

$$[P \rceil \text{.}$$

The empty word will be given the length zero:

(1) $[\Lambda \rceil = 0$

Evidently every letter has length 1:

(2) $[\xi \rceil = 1 \; (\xi \in A)$.

It is also clear from the definition of the length of a word that for all non-empty words P and Q the equality holds

(3) $[PQ \rceil = [P \rceil + [Q \rceil$.

On the strength of 6(2) and (1) it is true even in the case when one of the two words is empty.

8.1. <u>The equality (3) holds for any words P and Q.</u>
On the strength of 7.1, (2) and(3), the following lemmae hold.

8.2. <u>Every non-empty word P in the alphabet A can be represented in the form ξQ, where $\xi \in A$, and Q is a word in A such that</u>

(4) $\quad [Q\partial = [P\partial - 1$.

8.3. <u>Every non-empty word P in the alphabet A can be represented in the form $Q\xi$ where $\xi \in A$ and Q is a word in A such that</u> (4) <u>holds.</u>

The following method of proof of general assertions about words in a given alphabet A can be based on lemma 8.2.

Suppose we want to prove that all words in A possess a certain property $\mathcal{P}$. We will prove for this the following two assertions:

1) The empty word possesses the property $\mathcal{P}$.
2) If some word Q in A possesses property $\mathcal{P}$,
then whatever letter ξ in the alphabet A, the word ξQ also possesses the property $\mathcal{P}$.

Then we may assert that every word in A possesses the property $\mathcal{P}$.

In this fashion, on the strength of 1), we can assert that every word of length Q in A possesses property $\mathcal{P}$, because Λ is the only such word. We will assume that we also have demonstrated that every word of length $k - 1$, for $k > 0$, possesses property $\mathcal{P}$. We consider then some word P in A of length k. It is empty, and therefore

$$P = \xi Q$$

for a certain letter ξ of the alphabet A and a certain word Q in A satisfying condition

(4) $\quad$ [8.2]. Because $[P\partial = k$, we have
$[Q\partial = k - 1 \qquad [(4)]$

and this means that by hypothesis, Q possesses the property $\mathcal{P}$. Consequently, P possesses it [(5), 2]. By this we have proved that all words of length k in A possess property A, as soon as every word of length k - 1 in A possesses it. This permits, step by step, determination sequentially that property $\mathcal{P}$ is possessed by all words of length 1 in A, that it is possessed by all words of length 2 in A, etc. up to a word of arbitrary given length n inclusive. Consequently, property $\wp$ is possessed by every word in A, which was required to be proved.

The basis of just such methods of proof of general assertions about words in a given alphabet is a variant of the method of mathematical induction*. We will call this the <u>method of left induction in A</u>. Completely analogous, the method of right induction in A, obtained by replacing ξQ in assertion 2) by $Q\xi$, is based on a similar method with the aid of lemma 8.3.

---

*We recall in connection with this that the spreading opinion about this, that the basis of the method of mathematical induction certainly requires a basic "axiom of mathematical induction", in our opinion, deeply wrong.